HEFS-1.1.1 Release Notes

Release Date: 4/29/2014
Release Type: Scheduled
HEFS Build: 1.1.1
Build and Package Date: 4/29/2014
Tested against FEWS Binary: 2013.01 build 45343, patched from 42428

Introduction

This document contains release notes for HEFS-1.1.1.

- **IMPORTANT:** This release of HEFS-1.1.1 assumes you have already upgraded your CHPS to version 4.0.1 or later.

- If your RFC has already updated to HEFS-1.0.2, MEFPPE and EnsPostPE have not changed, and parameters do not need to be re-estimated.

- As part of this release, all configuration associated with GFS forecast source should be removed. The GFS forecast source was associated with the 1997 frozen version of the GFS which is no longer operationally used. See "Removing GFS from MEFP Workflows [BUG]" in the Fixes section.

- Do not use the raw version of the climatology forecast source in MEFP (turned on by setting the useResampledClimatology run-file property to "false"). There are two reasons. First, MEFP generated raw climatology is not true raw climatology for temperature; specifically, the resultant 6-hour FMAT time series generated based on MEFP output will not reflect the actual 6-hour historical MAT data due to application of diurnal patterns. Second, the MEFP raw climatology functionality duplicates functionality already available in CHPS via transformations. As such, existing transformations configured for ESP forecasts should be used, including sample transformations (typically named "SampleESP…") to acquire the raw climatology data, and a merge transformation to append that data on the end of the output generated using MEFP. For more information, see Section 4.4.9.2 of the *MEFP User's Manual* and Section 4 of the *MEFP Configuration Guide: Forecast Components*.

- As part of this release of HEFS, a new release of Graphics Generator is included. This release includes features not compatible with the older release of Graphics Generator currently available by default in CHPS. As such, any referenced templates or product templates saved using the HEFS 1.1.1 version of Graphics Generator delivered with this software will not be readable using the older version in CHPS. For that reason, it is not recommended that the HEFS 1.1.1 version of Graphics Generator be used to edit templates unless it is the only version of Graphics Generator used for that purpose.

- Below are two tables of the fixes and enhancements in this release. Following that is a list of changed documentation. Afterwards, the fixes and then enhancements are described in greater detail.

- The HEFS Test RFCs are expected to test existing, fixed, and new HEFS functionality which they reported and requested. The Test Manual provided in this release further specifies what testing is expected from each RFC and contains the test procedures for many tests.

Fixes

FogBugz ID	Reported By	Title
889	CNRFC	CFSv2 Precip too high
1091	CNRFC	GDS Exception Error
1239	CNRFC	EnsPost Hindcast Heap Memory Usage
1243	ABRFC	MEFPPE Issues with HEFS 1.0.1
1275	CNRFC	EnsPost Config Error: index >=size
1344	Hank Herr (OHD)	Removing GFS from MEFP Workflows
1369	CNRFC	MEFP Grid-Basin relationship
1384	NERFC	MAT to TAMN transform problem

Enhancements

FogBugz ID	Requested By	Title
1253	CNRFC	MEFP RFC Observations
1265	CNRFC	EnsPost Use of Location and Module Instance Sets
1367	CBRFC	Add output of diagnostics graphics to MEFPPE
1371	Hank Herr (OHD)	Restricting Permissions on HEFS Workflows and Buttons

Documentation

The following pieces of documentation have been modified since the last release and can be found in the 'documentation' directory at the root of the package. All the HEFS documentation may be found online at http://www.nws.noaa.gov/oh/hrl/general/indexdoc.htm.

- *Modified:*
  - EnsPost Configuration Guide: Forecast Components
  - EnsPost Users Manual
  - EVS Users Manual
  - HEFS Overview and Getting Started
  - HEFS Test Manual 1.11
  - Hindcasting Guide
  - MEFP Configuration Guide: Data Ingest
  - MEFP Configuration Guide: Forecast Components
  - MEFPPE Configuration Guide
  - MEFP User’s Manual

- *New:*
  - Hindcasting Robot User Manual

Notes

Fix: FogBugz 889 - CFSv2 Precip too high

Description

CHPS is displaying twice the precip amount in the CFSv2 grib file.

Cause

Units conversion is incorrect in CFSv2 import configuration.

Fix

Correct units conversion in. ../Config/UnitConversionsFiles/ImportMEFPCFSv2Units.xml

Notes

Fix: FogBugz 1091 - GDS Exception Error

Description

Error creating database connection, program will not launch.

Cause

The part of the code that downloads grids first places them under /tmp. It then compares the downloaded file with previously downloaded files and, if they are different, it moves the tmp file to the permanent location, overwriting any existing files (this is done to avoid updating the file's last modified time when the file wasn't modified, which in turn would cause the MEFPPE interface to display too many yellow warning icons in its summary tables). The problem is that, if they are the same, it would do nothing with the /tmp file; it should delete it, but it doesn't.

Fix

Properly delete /tmp files.

Notes

Fix: FogBugz 1239 - EnsPost Hindcast Heap Memory Usage

Description

Running EnsPost in batch mode (hindcast) slows dramatically and sometimes crashes.

Cause

Log events were created for each EnsPost run but the memory was not freed by the HEFS Adapter after the logs were used and released. The log grows to consume all memory resources.

Fix

Update the HEFS Adapter to garbage collect diagnostic objects.

Notes

Fix: FogBugz 1243 - MEFPPE Issues with HEFS 1.0.1

Description

4 issues reported with HEFS 1.0.1 MEFPPE:

1. We can only pull up the export MAP/MAT/TMAX/TMIN window once before it crashes CHPS.

2. Doing a large copy of the RFC SHEF code to other ids causes the a 'do you want to quit the application?' error.

3. While doing MAP estimations, the random estimation icons remain yellow.

4. When doing a run-all CHPS would not always remove the old progress bars for already completed locations. These show up on the top and are never removed.

Other small enhancements made to MEFPPE will also be outlined below in the Notes.

Causes

1. There is a memory issue with how the getTimeSeriesHeaders() method in the Deltares PI-service runs. Using an ABRFC parameter estimation stand-alone, which included historical data for 100+ points, the method would require about 1.5 GB of memory while testing in HEFS 1.0.2. I assume this is the same for HEFS 1.0.1, and it indicates the method must be reading in the entire time series in order to acquire the headers.

2. The copy-to-all option for the RFC SHEF code column was accidentally removed.
3. This is a symptom of a more general problem of updating the icons of the Location Summary Panel was when a run-all job is performed.

4. There are multiple places in the code where if an exception occurs during parameter estimation, a progress bar would not be appropriately removed, causing left-over progress bars to stick around for the next segment.

Fixes

1. No fix is planned, but a work-around is to break the parameter estimation stand-alone into multiple stand-alones, each intended to access only a subset of the MEFP locations. To do so, create a copy (or copies) of the base stand-alone and modify the PiServiceConfigFiles/MEFPPE.xml for each stand-alone so that only some of the time series are returned.

2. The copy-to-all option has been restored.

3. The icons now update appropriately and an hourglass icon to indicate the parameters for a point are being estimated has been added.

4. Progress bars are now removed.

Notes

A default button has been added in the canonical events panel to reset events to their delivered, default values.

A backup has been added to the run-time-info system file so it can be recovered if it becomes corrupt.

A button to allow for loading estimation options used for estimating a specific set of parameters has been added to the Estimation Panel.

Undo and redo buttons have been added to the Estimation Options Subpanel within the Estimation Panel to allow for undoing and redoing changes to the estimation options.

Fix: FogBugz 1275 - EnsPost Config Error: index >=size

Description

When attempting to run EnsPost, get an error message ‘index >= size’.

Cause

The code was attempting to apply the error model to an empty ensemble when the timestep was already 24 hours, resulting in no aggregation being done.

Fix

The code will not apply the error model to an empty ensemble when the timestep is already 24 hours.

Notes

Fix: FogBugz 1344 - Removing GFS from MEFP Workflows

Description

GFS forecast source is removed from workflows, leaving the forecast sources RFC, GEFS, and CFSv2 as is.

Cause

GFS has been discontinued.

Fix

There is a long list of files to be edited, see FogBugz 1344 for the complete list.

http://schuylkill.nws.noaa.gov:7069/default.asp?1344

Notes

The files changes must be done at each RFC. Besides those files listed in FogBugz 1344, other RFC created and customized configuration files may have GFS references that also need to be removed.

Fix: FogBugz 1369 - MEFP Grid-Basin relationship

Description

Median ensemble temperatures for a basin were 6-10 degrees above our deterministic (best guess) forecast.

Cause

To determine the closest grid cell during parameter estimation, MEFPPE rounds the basin location coordinates. The FEWS spatial interpolation transformation 'closestDistance' is not equivalent to rounding.

Fix

Change the MEFPPE centroid so it uses the same grid cell as the FEWS spatial interpolation transformation 'closestDistance'.

Notes

Fix: FogBugz 1384 - MAT to TAMN Transform Problem

Description

The current step plot indicates that each TMIN value is recorded at the end of the 12Z-12Z 24-hour period to which it applies (i.e. each blue flat line step ends at the blue scatter point triangle).

Cause

The 24-hour TMIN values are recorded at the beginning of the 12Z-12Z 24-hour period (TMAX is recorded at the end).

Fix

The diagnostic plot is fixed so that the flat line steps will start at each scatter point for TMIN data. The TMAX plot will be unchanged.

Notes

Enhancement: FogBugz 1253 - MEFP RFC Observations

Description

Users must import the same precipitation observations twice in two different formats. During MEFP precipitation parameter estimation, RFC forecast and observation data is loaded. Is there a way to utilize the datacard files already imported into the CHPS database as RFC observations rather than having to reformat the datacards into the required format?

Cause

The historical data used for parameter estimation is a combination of the datacard data and RFC observations; the historical datacard data has priority and RFC obs is only used to fill in the gaps and extend the data. However, if a forecast source, such as RFC, provides its own observed data, that data is always used for parameter estimation instead of the historical data. Furthermore, MEFPPE requires both observed and forecast data for the RFC forecast source. Thus, the datacard data plays no role in RFC parameter estimation.

Fix

The RFC observed data is not required by MEFPPE for the RFC source. If it is not provided or is empty or all missing, then the regular historical datacard data is used. If it is provided and *any* data is not missing, then it is assumed that the observations for the RFC source are fully specified and specifically designed to be used with the RFC forecasts; therefore, the datacard data will not be used. In this case, it will not be merged with the datacard data, because it is not clear which would receive priority, datacard data or RFC obs, and the priority could vary by RFC. Thus a user must either provide ALL RFC observations in the obs file or no observations at all.

Notes

Enhancement: FogBugz 1265 - EnsPost Use of Location and Module Instance Sets

Description

EnsPost requires a separate .xml file per location. Is there a way to utilize LocationSets and ModuleInstanceSets so that the number of xml files can be greatly reduced?

Cause

EnsPost, during development, was viewed as equivalent to SACSMA or SNOW-17, in that it would need to be configured one per segment/forecast point.

Fix

EnsPost was enhanced so it can run in a loop for a group of locations.

Notes

Enhancement: FogBugz 1367 - Add output of diagnostics graphics to MEFPPE

Description

The only way to save MEFP parameter diagnostics is to open the SA, view an image and save them one at a time. This is not practical when needing to save over 600 images.

Cause

Working as designed.

Fix

Diagnostic images can be batch created.

Notes

1. In MEFPPE, select a single location for which to load parameters and view the parameter diagnostic (block plot) that you would like to view for many locations.

2. Select the locations for which to generate that plot in the location table of the Estimation panel.

3. In the Diagnostic Panel, click on a new button to reproduce the diagnostics for all of those selected locations. A dialog will open asking for an output directory and output file name prefix.

4. In the dialog that opens, select an output directory and specify a file name prefix for every image file output.

5. Click OK when done.

In alphabetical order, MEFPPE will load the parameters for each selected location, use the information in the diagnostic panel to generate a diagnostic image of the same size as the current parameter diagnostic panel, and output that image to the file:

```
<output directory>/<prefix>.<locationId>.<parameterId>.png
```

Since the run time for such an action will be long, a progress dialog will be used to show status.

Enhancement: FogBugz 1371 - Restricting Permissions on HEFS Workflows and Buttons

Description

How do you restrict permissions of the HEFS workflows and buttons?

Cause

Fix

Instructions are given in FogBugz 1371:

http://schuylkill.nws.noaa.gov:7069/default.asp?1371

Notes

Notes for EVS Bug Fixes and Enhancements

Fix: Incorrect addition of default Aggregation Unit (AUs)

Description

In the Aggregation window of the GUI, copying a default Aggregation Unit (AU) and then attempting to delete that AU, failed to remove the selected AU. Specifically, upon deleting the default AU, a new default AU was immediately added to the list of AUs. Thus, only fully-defined AUs could be deleted from the list.

Cause

In order to avoid the addition of duplicate AUs, the addDefaultUnits method of AggregationA.java checks for existing AUs with common parameter values. However, it was only checking for fully-defined AUs that would lead to duplication, and not for default AUs with the same candidate Verification Units (VUs).

Fix

Fixed the addDefaultUnits method of AggregationA.java to check for default AUs with the same candidate VUs before adding new default AUs.

Notes

Activated the same code by copying a default AU and then deleting the AU. This resulted in the AU being removed correctly.

Fix: Misleading error message on setting incorrect weights

Description

Weighted aggregation of two of more VUs requires the weights associated with each VU to be defined in the Aggregation window. The weights must sum to 1.0 and an error message is encountered when the weights do not sum to 1.0. However, when an AU has not been previously saved, the error hierarchy leads to a misleading, generic, error being thrown, which fails to identify the lack of unity in the weights as the trigger.

Cause

The catch block of the saveData method in AggregationA.java catches an error in the weights but, before throwing the error, attempts to reset the original weights. Since this attempt can fail, a more generic error may be thrown, triggering an exit of the saveData method before the original error can be thrown.

Fix

The catch block of the saveData method was updated. The code that attempts to reset the default weights is now placed in a try/catch block. If this fails, the error is caught, and the original error (highlighting weights that do not sum to unity) is correctly thrown. A failure to reset the original weights may occur if no valid weights were previously defined.

Notes

Setting incorrect weights on first saving an AU now results in the correct error message being thrown, instead of a more generic message, indicating to the user that the problem originates in the weights not summing to unity.

Fix: Failure to save a manually entered forecast file data source

Description

When attempting to save a manually entered path to a forecast file data source, the path was not properly saved unless an earlier path existed. In contrast, when an earlier path existed or the file dialog was used to populate the input, the path was properly saved.

Cause

An if-clause in the saveLocalData method of VerificationA.java was poorly composed. This resulted in a failure to catch and handle the above scenario (manually entered forecast file data source, without a prior data source defined), namely by saving the manually entered path to the forecast file data source.

Fix

Updated the if-clause in VerificationA.java to properly handle the scenario where a forecast file data source is manually defined for the first time.

Notes

Checked all possible combinations for defining the forecast file data source.

Fix: Failure to re-populate the list of a searchable combo-box

Description

A searchable combo-box allows for a list of items in the combo box to be re-populated with matching search text. However, upon finding an exact match, the list was re-populated with a single item and the search text needed to be deleted in order to restore the full (original) list. Also, when finding no matches, the combo box was populated with an empty list, rather than providing the full original list.

Cause

Two bugs were found in the SearchableComboBox.java, the first concerned with an exact match (list with one item) and the second concerned with an empty list (no matches). In both cases, the list associated with the combo-box was not being re-populated with the original items.

Fix

Corrected two bugs in the SearchableComboBox.java to ensure the combo-box list is re-populated with the original list when either a single item is found or no items are found.

Notes

Manually checked all keyboard and mouse interactions with the SearchableComboBox to ensure that: 1) the full list is returned when a single item is found or no items are found from the search text; and 2) a reduced list (based on the matching text) is returned in all other cases.

Fix: Failure to warn of permission errors when saving a project file

Description

When encountering a file permission error on writing an EVS project file, the resulting IOException was output to the command line only and failed to display in the GUI exception handler.

Cause

The writeObjects method of the EVSMainWindow.java class failed to throw an IOException, instead catching all exceptions and printing to standard out.

Fix

The writeObjects method of the EVSMainWindow.java class was modified to propagate all IOExceptions upwards, allowing for proper handling by calling methods, such as displaying the IOException in the GUI exception handler.

Notes

Recreated the problem by changing the file permission on an EVS project file to read only. Prior to the bug fix, any attempt to save the EVS project file would result in a "silent" error, printed to standard out only. Following the fix, the IOException is properly displayed in the GUI exception handler.

Fix: Disappearing modal dialogs

Description

Under some (inconsistent) conditions, a modal dialog would disappear behind other open windows, blocking further input to the EVS until the modal dialog re-appeared through trial-and-error minimization/maximization of other open windows.

Cause

The parent frame of the modal dialog was not being set properly on construction. Specifically, due to the order of construction of the modal dialogs before the main EVS main window, a null parent frame was passed to the constructors of the modal dialogs.

Fix

Changed the order of construction of the modal dialogs to follow the construction of the EVS main window. The modal dialogs are now constructed with a non-null parent frame.

Notes

Tested the modal dialogs under Windows and Linux. No further blocking of modal dialogs occurred.

Fix: Incorrect re-weighting of VUs associated with an AU

Description

Upon changing the weights of a VU associated with an AU in AggregationA.java, the remaining weights are updated accordingly (e.g. to re-assign equal weights when VUs are added or removed). The updates are coordinated by the setValueAt method of the table model associated with the table of VUs, which is called upon a change in the selection of a VU or a change in weight. However, the checks were not conditional upon the VU being flagged for inclusion or on the updated value of the entry passed to setValueAt. This led to incorrect updates to the weights upon editing unchecked VUs or when the current VU was being deselected (because the status was checked against the existing selection, not the updated selection).

Cause

Failure to properly check the status of the VUs and their associated weights in the setValueAt method of the table model associated with the table of VUs in AggregationA.java.

Fix

Updated the setValueAt method of AggregationA.java to properly check the status of the VUs and their associated weights upon adding or removing VUs or changing weights.

Notes

Tested the re-weighting of VUs upon adding or removing VUs and when the edited weight belongs to a VU that is checked versus unchecked. All scenarios performed as expected.

Fix: Failure to fully clear and reset the GUI on creating a new EVS project

Description

On creating a new EVS project, the GUI was not fully cleared, potentially leaving results from a prior open project. In addition, the GUI failed to return to the first window in the verification stage.

Cause

The newProject method of EVSMainWindow.java failed to clear all prior data and reset the GUI to the first window of the verification stage.

Fix

Updated the newProject method of EVSMainWindow.java to clear all prior data and reset the GUI to the first window of the verification stage.

Notes

Tested by generating results for one EVS project and then creating a new EVS project. On creating the new project, all data from the old project were fully cleared and the GUI returned to the first window in the verification stage.

Enhancement: Extend list of observed file types displayed in file dialog

Description

By default, only selected file extensions were identified as "observed" data files in the file dialog. The default file extensions did not include .MAP06 or .MAT, for example. In order to display these files in the dialog, it was necessary to select "All Files" in the drop-down menu provided in the file dialog. This was potentially confusing, as it may not be apparent that the file dialog is showing only a subset of the available files, based on the default extensions.

Cause

Limited list of default file extensions for observed data files. In particular, the default file types did not include MAT, MAP06 and MAP01, which are relatively common.

Fix

Added .MAP06, .MAP01, and .MAT to the list of default file types in the file_types.xml file of the nonsrc/parameterfiles directory.

Notes

Checked that the additional default file types were correctly displayed in the file dialog when choosing observed data files.

Enhancement: Handling of aggregation weights in the GUI

Description

When setting the weights for spatial aggregation of several VUs, there were several minor constraints in the aggregation window of the GUI that interrupted smooth user interaction.

Cause

When editing the weight for one VU in an AU comprising only two units, the weight of the remaining unit was not automatically updated, based on the requirement for the total weights to sum to 1.0. In addition, the editing cell in the table was not automatically closed upon completion, leaving the editor open when selecting other VUs. Finally, when the weights did not sum to 1.0, the error message was unclear about which AUs (in the presence of multiple units) failed to sum to 1.0.

Fix

Updated AggregationA.java to allow for the second of a pair of weights to be updated automatically when the AU comprises only two VUs and the first of the pair of weights is edited. The automatic update ensures that the weights sum to 1.0, as required. Enforced closure of all open cell editors in the table of weights in AggregationA.java when switching between VUs or saving the project. Improved the warning message when weights do not sum to 1.0, identifying the first AU for which the weights are incorrect.

Notes

Checked all enhancements manually by interacting with the GUI for different scenarios of the aggregation weights.

Enhancement: Support reading and writing of EVS pairs in compressed format

Description

The EVS pairs are written in an uncompressed ASCII XML format. In order to conserve disk space, the option for writing (and reading) of the EVS pairs in a compressed (gzip) format is useful.

Cause

Lack of functionality for writing and reading of the EVS paired files in a compressed (gzip) format leads to inefficient use of disk space.

Fix

Added optional functionality to read and write the EVS paired files (both raw and conditional) in a compressed gzip format. The default remains to write the pairs in an uncompressed format. The option to write the pairs in a compressed gzip format is accessed through the output options dialog in the first verification window. The output options dialog is accessed via the “More” button in “2d. Select location for output data”. Updated the I/O of the EVS project file to store the preference for compressed output. The option appears under the <write_gzip_pairs> tag of the <paired_data> block) and applies to each VU separately.

Notes

Checked that the compressed files were written and read correctly and that the option was written and read correctly to/from the EVS project file.

Enhancement: Dismissing multiple warnings with a single action

Description

Upon deleting multiple VUs, several warning messages may be thrown if there are dependencies between VUs. Rather than scrolling through all messages, the option to ignore all warnings would be useful.

Cause

The inability to accept all warnings when deleting multiple VUs.

Fix

Added an option in VerificationA.java to accept all warnings when deleting multiple VUs that have dependencies with other VUs (not all of which may be deleted). Each warning dialog now comprises an option “Yes to all” to accept all subsequent warnings and proceed.

Notes

Manually tested all three options, namely to cancel the delete, accept the current warning message and proceed (possibly to other warning messages), and to accept all warning messages and proceed to delete. All options worked as anticipated.

Enhancement: Support reading of Fast Infoset binary XML files

Description

Fast Infoset XML provides a binary interface to XML that allows for smaller file sizes and more efficient processing times.

Cause

Inability to process Fast Infoset binary XML files for the ensemble forecasts and observations.

Fix

Created a new reader, FastInfosetXMLIO.java, to support reading of Fast Infoset binary PI-XML for both ensemble forecasts and observations. The new reader relies on library methods in ohdcommonchps.jar and two Deltares libraries, Delft_Util.jar and Delft-PI.jar.

Notes

Conducted performance testing of read times for ASCII PI-XML and Fast Infoset binary XML. The read times and disk-space footprint for the Fast Infoset binary XML files were significantly smaller than ASCII PI-XML.

Enhancement: Support file filter when reading forecast directory

Description

When specifying a directory from which to read forecast files, any non-forecast files within that directory are processed as forecast files, which leads to an I/O exception. This is expected behaviour, but limiting when the forecasts are archived in directories with mixed files or subdirectories.

Cause

Inability to process a directory containing mixed forecast files by listing and processing only those files that match a prescribed filter.

Fix

Implemented a file filter to screen a directory of forecast files and process only those files that match a prescribed pattern. This only applies when a directory is selected as the forecast data source, and the filter must be defined explicitly (the default is to no filter). Added a new class, Filter.java, to filter the files, and updated the FileDataSource.java to store the filter. Also updated the ProjectFileIO.java to support reading/writing of the new filter option and MoreInputDialog.java to allow the filter to be entered in the GUI. Finally, VerificationA.java was updated to accommodate saving of the filter information.

Notes

Tested the default behavior (no filter) and the new filter option. The default behavior was maintained and the filter worked as expected, screening only those files that matched a prescribed pattern. The filter option was preserved in the GUI and in the project file, as expected.